

CES-35 – Redes de Computadores e Internet

Laboratório 1: Conhecendo protocolos - Wireshark

Nome: Daniel Araujo Cavassani (COMP 25)

Data: 26/08/2024

1. Captura de pacotes

- **Ação:** Iniciar captura no Wireshark após selecionar a interface de rede com acesso à internet.

2. Acesso ao site do Kurose

- **Ação:** Acessar <http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html> no navegador e parar a captura.
- **Arquivo de captura salvo em formato pcapng:** Sim

3. Perguntas Gerais

3.A) Quais destes protocolos aparecem na lista de pacotes: TCP, QUIC, HTTP, DNS, UDP, TLS?

- Resposta: TCP, QUIC, HTTP, DNS, UDP, TLS

3.B) Quanto tempo transcorreu desde o envio do HTTP GET até o recebimento do HTTP OK?

- Tempo decorrido: 126.654 ms

3.C) Qual a utilidade dos campos **User-Agent (HTTP GET) e **Server** (HTTP OK)?**

- Resposta:
 - O campo User-Agent no HTTP GET foi capturado como:

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.0.0 Safari/537.36\r\n

Esse campo é utilizado para informar ao servidor detalhes sobre o navegador e o sistema operacional do cliente. No caso, ele indica que o cliente está utilizando o navegador Google Chrome, versão 128.0.0.0, no sistema operacional Windows 10 (64 bits). Além disso, o navegador usa o motor de renderização AppleWebKit, comum ao Safari. Essa informação é importante para que o servidor adapte a resposta ao cliente, garantindo compatibilidade e otimização da exibição do conteúdo de acordo com as especificações do navegador e sistema operacional.

- Server: O campo Server na resposta HTTP OK foi capturado como:

Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0.11 Perl/v5.16.3\r\n

O campo Server informa ao cliente qual software está sendo utilizado no servidor. No caso, o servidor está rodando o Apache/2.4.6 em um sistema CentOS, com suporte a OpenSSL/1.0.2k-fips para criptografia segura, PHP/7.4.33 para execução de scripts PHP, e mod_perl/2.0.11 para processar scripts Perl. Essas informações ajudam a entender a infraestrutura do servidor, o que pode ser útil para depuração e compatibilidade de sistemas.

3.D) Tamanho dos cabeçalhos e dados úteis da resposta HTTP OK:

- Cabeçalho de Aplicação (HTTP): 357 bytes
- Cabeçalho de Transporte (TCP): 20 bytes
- Cabeçalho de Rede (IP): 20 bytes
- Cabeçalho de Enlace (Ethernet): 14 bytes
- Total de bytes dedicados aos cabeçalhos: 411 bytes
- Dados úteis: 81 bytes
- Porcentagem de dados úteis: 16.46%

3.E) Prints das mensagens HTTP GET e HTTP OK:

http						
No.	Time	Source	Destination	Protocol	Length	Info
4844	01:31:57,760175	192.168.51.141	128.119.245.12	HTTP	574	GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1
5127	01:31:57,895300	128.119.245.12	192.168.51.141	HTTP	492	HTTP/1.1 200 OK (text/html)

- HTTP GET

```
> Frame 4844: 574 bytes on wire (4592 bits), 574 bytes captured (4592 bits) on interface \Device\NPF_{3D18F8DC-D80A-49FC-89BF-EF4290393961}, id 0
> Ethernet II, Src: b0:25:aa:42:3e:5b (b0:25:aa:42:3e:5b), Dst: Dell 4e:0e:73 (18:66:da:4e:0e:73)
> Internet Protocol Version 4, Src: 192.168.51.141, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 51907, Dst Port: 80, Seq: 1, Ack: 1, Len: 520
Hypertext Transfer Protocol
  GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1\r\n
    [Expert Info (Chat/Sequence): GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1\r\n]
    [GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1\r\n]
    [Severity level: Chat]
    [Group: Sequence]
    Request Method: GET
    Request URI: /wireshark-labs/INTRO-wireshark-file1.html
    Request Version: HTTP/1.1
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Cache-Control: max-age=0\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.0.0 Safari/537.36\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7\r\n
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: en-US,en;q=0.9,pt-BR;q=0.8,pt;q=0.7\r\n
    \r\n
    [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html]
    [HTTP request 1/1]
    [Response in frame: 5127]
```

- HTTP OK

```
> Frame 5127: 492 bytes on wire (3936 bits), 492 bytes captured (3936 bits) on interface \Device\NPF_{3D18F8DC-D80A-49FC-89BF-EF4290393961}, id 0
> Ethernet II, Src: Dell 4e:0e:73 (18:66:da:4e:0e:73), Dst: b0:25:aa:42:3e:5b (b0:25:aa:42:3e:5b)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.51.141
> Transmission Control Protocol, Src Port: 80, Dst Port: 51907, Seq: 1, Ack: 521, Len: 438
Hypertext Transfer Protocol
  HTTP/1.1 200 OK\r\n
    [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
    [HTTP/1.1 200 OK\r\n]
    [Severity level: Chat]
    [Group: Sequence]
    Response Version: HTTP/1.1
    Status Code: 200
    [Status Code Description: OK]
    Response Phrase: OK
    Date: Tue, 27 Aug 2024 01:31:57 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0.11 Perl/v5.16.3\r\n
    Last-Modified: Mon, 26 Aug 2024 05:59:01 GMT\r\n
    ETag: "51-6208fd25bd448"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 81\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    \r\n
    [HTTP response 1/1]
    [Time since request: 0.135125000 seconds]
    [Request in frame: 4844]
    [Request URI: http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html]
    File Data: 81 bytes
  Line-based text data: text/html (3 lines)
```

3.F) Explicação sobre a mensagem HTTP 1.1/304 Not Modified e solução:

- Resposta:
 - Filtro aplicado: No campo de filtro do Wireshark, usei o seguinte filtro para capturar apenas as mensagens relacionadas ao IP do site do Kurose: ip.addr == 128.119.245.12
 - Acesso à página: Após acessar novamente a URL http://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html, recebemos um pacote de resposta com o código HTTP 1.1 304 Not Modified.

- Explicação do código HTTP 304 Not Modified: O código 304 Not Modified indica que o recurso solicitado não foi modificado desde a última vez que o navegador fez a requisição. Isso ocorre porque o navegador enviou cabeçalhos de cache como If-Modified-Since ou If-None-Match, e o servidor constatou que a versão do conteúdo no cache do navegador ainda é válida. Como resultado, o servidor não transfere novamente os dados, economizando largura de banda.

- Como evitar o HTTP 304 Not Modified: Para evitar receber a resposta 304 e forçar o servidor a enviar o conteúdo completo novamente, é possível:
 - Esvaziar o cache do navegador: Limpar o cache força o navegador a solicitar uma nova versão do recurso.
 - Utilizar o modo anônimo/privado: Nesse modo, o navegador não utiliza cache e sempre solicita o conteúdo completo.
 - Desativar temporariamente o cache: Nas ferramentas de desenvolvedor do navegador, é possível desativar o cache, garantindo que o servidor envie novamente os dados completos.

4. Camada de Transporte

4.A) Qual é o número da porta de destino e de origem para o HTTP GET?

- Porta de origem: 51907
- Porta de destino: 80

4.B) Campos da camada TCP encontrados no pacote:

- Resposta:
 - Source Port (Porta de Origem): 51907
 - Destination Port (Porta de Destino): 80
 - TCP Segment Length: 520 bytes
 - Sequence Number (Número de Sequência): 1 (relative sequence number)
 - Acknowledgment Number (Número de Confirmação): 1 (relative acknowledgment number)
 - Header Length (Tamanho do Cabeçalho): 20 bytes (5 x 4 = 20 bytes)
 - Flags:
 - PSH (Push Flag)
 - ACK (Acknowledgment Flag)
 - Window Size (Tamanho da Janela): 512
 - Window Size Scaling Factor: 256
 - Checksum: 0xbdc6 (unverified)
 - Urgent Pointer: 0 (não utilizado)
 - TCP Payload: 520 bytes

4.C) Flags no 3-way handshake:

ipaddr == 128.119.245.12							
No.	Time	Source	Destination	Protocol	Length	Info	
4299	01:31:57,627066	192.168.51.141	128.119.245.12	TCP	66	51907 → 80	[SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
4300	01:31:57,627290	192.168.51.141	128.119.245.12	TCP	66	51908 → 80	[SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
4745	01:31:57,723732	192.168.51.141	128.119.245.12	TCP	66	51909 → 80	[SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
4842	01:31:57,759974	128.119.245.12	192.168.51.141	TCP	66	80 → 51907	[SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM WS=128
4843	01:31:57,760009	192.168.51.141	128.119.245.12	TCP	54	51907 → 80	[ACK] Seq=1 Ack=1 Win=131072 Len=0
4844	01:31:57,760175	192.168.51.141	128.119.245.12	HTTP	574	GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1	
5070	01:31:57,856932	128.119.245.12	192.168.51.141	TCP	66	80 → 51909	[SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM WS=128
5071	01:31:57,857036	192.168.51.141	128.119.245.12	TCP	54	51909 → 80	[ACK] Seq=1 Ack=1 Win=131072 Len=0
5126	01:31:57,894370	128.119.245.12	192.168.51.141	TCP	60	80 → 51907	[ACK] Seq=1 Ack=521 Win=30336 Len=0
5127	01:31:57,895300	128.119.245.12	192.168.51.141	HTTP	492	HTTP/1.1 200 OK (text/html)	
5170	01:31:57,944131	192.168.51.141	128.119.245.12	TCP	54	51907 → 80	[ACK] Seq=521 Ack=439 Win=130816 Len=0
5509	01:31:58,641870	192.168.51.141	128.119.245.12	TCP	66	[TCP Retransmission] 51908 → 80	[SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
5832	01:31:58,773933	128.119.245.12	192.168.51.141	TCP	66	80 → 51908	[SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM WS=128
5833	01:31:58,774021	192.168.51.141	128.119.245.12	TCP	54	51908 → 80	[ACK] Seq=1 Ack=1 Win=131072 Len=0
8230	01:32:02,898301	128.119.245.12	192.168.51.141	TCP	60	80 → 51907	[FIN, ACK] Seq=439 Ack=521 Win=30336 Len=0
8231	01:32:02,898341	192.168.51.141	128.119.245.12	TCP	54	51907 → 80	[ACK] Seq=521 Ack=440 Win=130816 Len=0

Primeiro pacote (SYN):

- Flag(s) de controle ligado(s): SYN
- Pacote: 4299

- **Descrição:** O primeiro pacote do handshake é enviado pelo cliente (192.168.51.141) para o servidor (128.119.245.12) com a flag **SYN** ativada, iniciando a conexão TCP.

Segundo pacote (SYN, ACK):

- **Flag(s) de controle ligado(s):** SYN, ACK
- **Pacote:** 4842
- **Descrição:** O segundo pacote é enviado pelo servidor (128.119.245.12) para o cliente (192.168.51.141) com as flags **SYN** e **ACK** ativadas, confirmando o recebimento do SYN do cliente e enviando o próprio SYN do servidor.

Terceiro pacote (ACK):

- **Flag(s) de controle ligado(s):** ACK
- **Pacote:** 4843
- **Descrição:** O terceiro pacote é enviado pelo cliente (192.168.51.141) para o servidor (128.119.245.12) com a flag **ACK** ativada, confirmando o recebimento do SYN do servidor e finalizando o handshake.

4.D) Portas envolvidas no HTTP OK:

- Porta de origem: 80
- Porta de destino: 51907

4.E) Desconexão após transferência da página:

No.	Time	Source	Destination	Protocol	Length	Info
4299	01:31:57,627066	192.168.51.141	128.119.245.12	TCP	66	51907 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
4300	01:31:57,627290	192.168.51.141	128.119.245.12	TCP	66	51908 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
4745	01:31:57,723732	192.168.51.141	128.119.245.12	TCP	66	51909 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
4842	01:31:57,759974	128.119.245.12	192.168.51.141	TCP	66	80 → 51907 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM WS=128
4843	01:31:57,768009	192.168.51.141	128.119.245.12	TCP	54	51907 → 80 [ACK] Seq=1 Ack=1 Win=131072 Len=0
4844	01:31:57,760175	192.168.51.141	128.119.245.12	HTTP	574	GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1
5070	01:31:57,856932	128.119.245.12	192.168.51.141	TCP	66	80 → 51909 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM WS=128
5071	01:31:57,857036	192.168.51.141	128.119.245.12	TCP	54	51909 → 80 [ACK] Seq=1 Ack=1 Win=131072 Len=0
5126	01:31:57,894370	128.119.245.12	192.168.51.141	TCP	60	80 → 51907 [ACK] Seq=1 Ack=521 Win=30336 Len=0
5127	01:31:57,895300	128.119.245.12	192.168.51.141	HTTP	492	HTTP/1.1 200 OK (text/html)
5170	01:31:57,944131	192.168.51.141	128.119.245.12	TCP	54	51907 → 80 [ACK] Seq=521 Ack=439 Win=130816 Len=0
5209	01:31:58,051170	192.168.51.141	128.119.245.12	TCP	66	80 → 51909 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 WS=256 SACK_PERM
5832	01:31:58,773933	128.119.245.12	192.168.51.141	TCP	66	80 → 51909 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1412 SACK_PERM WS=128
5833	01:31:58,774021	192.168.51.141	128.119.245.12	TCP	54	51908 → 80 [ACK] Seq=1 Ack=1 Win=131072 Len=0
8230	01:32:02,898301	128.119.245.12	192.168.51.141	TCP	60	80 → 51907 [FIN, ACK] Seq=439 Ack=521 Win=30336 Len=0
8231	01:32:02,898341	192.168.51.141	128.119.245.12	TCP	54	51907 → 80 [ACK] Seq=521 Ack=440 Win=130816 Len=0

Pacote nº 8230:

- **Instante de tempo:** 01:32:02,898301
- **Fonte:** 192.168.51.141 (porta 51907)
- **Destino:** 128.119.245.12 (porta 80)
- **Flags:** [FIN, ACK]

- **Descrição:** O cliente sinaliza que deseja encerrar a conexão com o servidor enviando o pacote com as flags FIN e ACK ativadas.

Pacote nº 8231:

- **Instante de tempo:** 01:32:02,898341
- **Fonte:** 128.119.245.12 (porta 80)
- **Destino:** 192.168.51.141 (porta 51907)
- **Flags:** [ACK]
- **Descrição:** O servidor responde ao pacote de FIN, ACK enviado pelo cliente com um pacote contendo apenas a flag ACK, confirmando o fechamento da conexão.

No contexto da desconexão TCP, o pacote 8230 é o que contém a flag FIN, iniciando o fechamento da conexão na porta 51907.

Em seguida, o servidor responde com um ACK (pacote 8231), encerrando a conexão conforme o protocolo TCP.

Portanto, a desconexão aconteceu na porta 51907, envolvendo os pacotes 8230 e 8231.

5. Endereços de rede

```
C:\Users\danie>ipconfig /all

Windows IP Configuration

Host Name . . . . . : DanxC-A72-Liv
Primary Dns Suffix . . . . . :
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
DNS Suffix Search List. . . . . : rede

Wireless LAN adapter Local Area Connection* 1:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter
Physical Address. . . . . : FC-B3-BC-4D-9E-06
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes

Wireless LAN adapter Local Area Connection* 10:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Microsoft Wi-Fi Direct Virtual Adapter #2
Physical Address. . . . . : FE-B3-BC-4D-9E-05
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . : rede
Description . . . . . : Realtek Gaming 2.5GbE Family Controller
Physical Address. . . . . : B0-25-AA-42-3E-5B
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IPv4 Address. . . . . : 192.168.51.141(Preferred)
Subnet Mask . . . . . : 255.255.254.0
Lease Obtained. . . . . : segunda-feira, 26 de agosto de 2024 13:40:35
Lease Expires . . . . . : terça-feira, 27 de agosto de 2024 00:40:36
Default Gateway . . . . . : 192.168.50.1
DHCP Server . . . . . : 192.168.50.1
DNS Servers . . . . . : 192.168.50.1
NetBIOS over Tcpip. . . . . : Enabled

Wireless LAN adapter Wi-Fi:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Intel(R) Wi-Fi 6 AX201 160MHz
Physical Address. . . . . : FC-B3-BC-4D-9E-05
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes

Ethernet adapter Bluetooth Network Connection:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
Description . . . . . : Bluetooth Device (Personal Area Network)
Physical Address. . . . . : FC-B3-BC-4D-9E-09
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
```

6. Camada de Rede

6.A) Endereços IP do gaia.cs.umass.edu e de seu computador:

- IP do gaia.cs.umass.edu: 128.119.245.12
- IP de seu computador: 192.168.51.141

6.B) O campo inet no ifconfig/ipconfig corresponde ao IP mostrado pelo Wireshark?

- Resposta: Sim

6.C) Campos da camada IP encontrados no pacote:

- Resposta:
 - **Version:** 4
 - **Header Length:** 20 bytes
 - **Differentiated Services Field:** 0x00 (DSCP: CS0, ECN: Not-ECT)
 - **Total Length:** 560
 - **Identification:** 0x8dda (36314)
 - **Flags:** 0x2 (Don't fragment)
 - **Fragment Offset:** 0
 - **Time to Live (TTL):** 128
 - **Protocol:** TCP (6)
 - **Header Checksum:** 0x0000 (validation disabled)
 - **Header Checksum Status:** Unverified
 - **Source Address:** 192.168.51.141
 - **Destination Address:** 128.119.245.12

7. Camada de Enlace

7.A) Endereços MAC de origem e destino:

- **Endereço MAC de origem:** b0:25:aa:42:3e:5b
- **Endereço MAC de destino:** 18:66:da:4e:0e:73

Esses são os endereços de origem e destino observados na camada Ethernet II. O endereço de origem é o da nossa placa de rede, e o endereço de destino é o próximo salto (que pode ser o roteador).

7.B) Campos da camada MAC encontrados no pacote:

- **Destination MAC Address:** 18:66:da:4e:0e:73
- **Source MAC Address:** b0:25:aa:42:3e:5b
- **Type:** IPv4 (0x0800)

Esses são os principais campos encontrados na camada de enlace (Ethernet) para o pacote HTTP GET.

7.C) O endereço ether no ifconfig corresponde ao endereço de origem mostrado pelo Wireshark?

O Endereço MAC de origem mostrado no Wireshark é b0:25:aa:42:3e:5b.

No resultado do comando `ipconfig /all`, o Endereço Físico (ou Physical Address) da sua interface Ethernet também é b0:25:aa:42:3e:5b.

Portanto, sim, o campo ether mostrado no `ipconfig` corresponde ao endereço de origem mostrado no Wireshark. Isso confirma que o seu sistema operacional está usando corretamente o endereço de sua placa de rede para montar os pacotes que emite para a rede.

8. Traceroute

8.A) Quantos saltos foram necessários até chegar ao site?

- O traceroute (tracert no windows) mostra que foram necessários 25 saltos para chegar ao servidor gaia.cs.umass.edu (128.119.245.12).

8.B) Há algum salto com tempo menor que o anterior?

Sim, há passos em que o valor de tempo é menor que o anterior, por exemplo:

- **Salto 9 (114 ms) para Salto 8 (124 ms).**
- **Salto 18 (126 ms) para Salto 17 (130 ms).**

Isso pode acontecer por várias razões:

- **Rotas Assíncronas:** Os pacotes podem seguir rotas ligeiramente diferentes entre os saltos, resultando em tempos de resposta diferentes.
- **Congestionamento Temporário:** O congestionamento de rede em um salto pode causar um tempo de resposta maior, mas o congestionamento pode diminuir em saltos posteriores.
- **Otimização de Roteadores:** Alguns roteadores podem ser mais rápidos ao processar pacotes devido a configurações de cache, resultando em tempos menores nos saltos subsequentes.

8.C) Saída do traceroute:

```
C:\Users\danie>tracert gaia.cs.umass.edu

Tracing route to gaia.cs.umass.edu [128.119.245.12]
over a maximum of 30 hops:

  0  8 ms   9 ms   9 ms  192.168.50.1
  1  1 ms   <1 ms  1 ms  192.168.100.1
  2  4 ms   3 ms   2 ms  186-194-168-2.dynamic.grupocompunet.com.br [186.194.168.2]
  3  8 ms   3 ms   3 ms  100.127.1.1
  4  2 ms   *       *     172.29.12.169
  5  2 ms   2 ms   2 ms  200.220.128.105.nipcable.com [200.220.128.105]
  6  4 ms   4 ms   4 ms  198.18.40.17
  7  124 ms 114 ms 115 ms ae1255.0.edge2.jfk1.as7195.net [200.25.51.54]
  8  114 ms *     114 ms ae0.0.edge1.jfk1.as7195.net [200.25.51.236]
  9  *       *       *     Request timed out.
 10 114 ms 116 ms 118 ms be3362.ccr41.jfk02.atlas.cogentco.com [154.54.3.9]
 11 120 ms 119 ms 121 ms be3471.ccr31.bos01.atlas.cogentco.com [154.54.40.153]
 12 122 ms 122 ms 122 ms be2729.rcr51.orh01.atlas.cogentco.com [154.54.40.182]
 13 120 ms 135 ms 127 ms 38.104.218.14
 14 136 ms 132 ms 125 ms 69.16.0.8
 15 125 ms 124 ms 123 ms 69.16.1.0
 16 126 ms 124 ms 125 ms core2-rt-et-8-3-0.gw.umass.edu [192.80.83.113]
 17 157 ms 126 ms 126 ms n1-rt-1-1-et-10-0-0.gw.umass.edu [128.119.0.120]
 18 125 ms *     125 ms 128.119.7.74
 19 125 ms 126 ms 126 ms 128.119.7.66
 20 127 ms 126 ms 126 ms core1-rt-et-7-2-1.gw.umass.edu [128.119.0.217]
 21 128 ms 127 ms 131 ms n5-rt-1-1-xe-2-1-0.gw.umass.edu [128.119.3.33]
 22 125 ms 125 ms 128 ms cics-rt-xe-0-0-0.gw.umass.edu [128.119.3.32]
 23 126 ms 130 ms 127 ms nscslbbs1.cs.umass.edu [128.119.240.253]
 24 127 ms 126 ms 127 ms gaia.cs.umass.edu [128.119.245.12]

Trace complete.
```